

AMENDMENT TO THE CLAIMS:

Please add claim 11.

This listing of the claims replaces all prior versions, and listings, of claims in the application.

- 1 1. (Original) An interface between a master and one or more slave modules
- 2 comprising:
- 3 a master;
- 4 a slave having a set of addressable registers including egress mailbox registers,
- 5 ingress mailbox registers, and indirect access address registers;
- 6 a direct memory access (DMA) engine coupled to the master by a first bus and
- 7 to the slave by a second bus, with the second bus comprising:
- 8 a set of bi-directional data lines for transmitting data between the slave and the
- 9 DMA engine;
- 10 a set of master address lines for transmitting address data from the DMA
- 11 engine to the slave;
- 12 a master data strobe for strobing data;
- 13 a master read/write signal for indicating whether data is to be read from or
- 14 written to the slave;
- 15 a set of slave select signals for selecting one of a plurality of slaves connected
- 16 to the second bus;
- 17 a slave wait signal asserted by a slave to delay a data transfer;
- 18 a slave reset signal,
- 19 a clock output signal, and
- 20 a clock input signal;
- 21 where the DMA engine performs direct data transfers to the slave by asserting
- 22 a slave select signal to the slave and transferring data over the set of bi-directional data lines
- 23 to the slave egress or ingress data registers and performs indirect data transfers to slave
- 24 memory by writing address data over the set of bi-directional data lines to the indirect address
- 25 register of the slave and where the slave utilizes its own memory map and the address data to
- 26 transfer data between a location indicated by the address data and the DMA engine.

1 2. (Original) The interface of claim 1 where the DMA engine negotiates with
2 a slave to implement either an asynchronous, synchronous, or source synchronous data
3 transfer.

1 3. (Original) The interface of claim 1 where the DMA engine negotiates with
2 all slaves during reset to determine the maximum bus width available to transfer data.

1 4. (Original) The interface of claim 1 where:
2 the slave includes status register and message signal interrupt (MSI) register;
3 and
4 where the slave asserts a bit in the status register to indicate it is ready for a
5 transaction and where the DMA engine asserts a bit in the MSI register to indicate when a
6 transaction is complete.

1 5. (Original) A method for allowing a DMA engine to provide access to a
2 plurality of slave devices to multiple masters, the protocol, implemented by hardware and
3 software on the DMA engine, the master, and the slave devices, comprising the steps of:
4 to implement a direct message transfer to a slave device:
5 accessing a slave status register to read a direct message ready status bit which
6 is set when the slave is ready to transfer data;
7 transferring message data using the DMA engine and a slave mailbox register
8 if the direct message ready status bit is set;
9 setting an message transfer complete status interrupt at the slave to indicate
10 when the transfer of the message is complete; and
11 to implement an indirect data transfer to the memory space of a slave device:
12 accessing a slave status register to read an indirect message ready status bit
13 which is set when the slave is ready to transfer data;
14 transferring address data using the DMA engine and slave indirect address
15 mailbox register if the indirect message ready status bit is set;
16 setting an indirect transfer message interrupt bit at the slave to initiate the
17 indirect transfer;
18 transferring message data between the DMA engine and slave mailbox
19 registers if the indirect message ready status bit is set, where the slave utilizes its own

20 memory map and the address data to transfer data between a location indicated by the address
21 data and the DMA engine; and
22 setting an message transfer complete status interrupt at the slave to indicate
23 when the transfer of the message is complete.

1 6. (Original) The method of claim 5 further comprising the step of:
2 negotiating with all the slaves to implement either an asynchronous,
3 synchronous, or source synchronous data transfer.

1 7. (Original) The method of claim 5 further comprising the step of:
2 starting the bus upon reset at a fixed bus-width and then negotiating with all
3 the slaves to implement acceptable bus bit-width.

1 8. (Original) A system for allowing a DMA engine to provide access to a
2 plurality of slave devices to multiple masters, the protocol, implemented by hardware and
3 software on the DMA engine, the master, and the slave devices, said system comprising:
4 means for implementing a direct message transfer to a slave device including:
5 means for accessing a slave status register to read a direct message ready
6 status bit which is set when the slave is ready to transfer data;
7 means for transferring message data using the DMA engine and a slave
8 mailbox register if the direct message ready status bit is set;
9 means for setting an message transfer complete status interrupt at the slave to
10 indicate when the transfer of the message is complete; and
11 means for implement an indirect data transfer to the memory space of a slave
12 device including:
13 means for accessing a slave status register to read an indirect message ready
14 status bit which is set when the slave is ready to transfer data;
15 means for transferring address data using the DMA engine and slave indirect
16 address mailbox register if the indirect message ready status bit is set;
17 means for setting an indirect transfer message interrupt bit at the slave to
18 initiate the indirect transfer;
19 means for transferring message data between the DMA engine and slave
20 mailbox registers if the indirect message ready status bit is set, where the slave utilizes its

21 own memory map and the address data to transfer data between a location indicated by the
22 address data and the DMA engine; and
23 means for setting an message transfer complete status interrupt at the slave to
24 indicate when the transfer of the message is complete.

1 9. (Original) The system of claim 8 further comprising:
2 means for negotiating with all the slaves to implement either an asynchronous,
3 synchronous, or source synchronous data transfer.

1 10. (Original) The system of claim 8 further comprising:
2 means for starting the bus upon reset at a fixed bus-width and then negotiating
3 with all the slaves to implement acceptable bus bit-width.

1

1 11. (New) A slave device for allowing a DMA engine to provide access by multiple
2 masters, said slave device comprising:
3 a slave status register accessed to read a direct message ready status bit which is set
4 when the slave is ready to transfer data directly and accessed to read an indirect message ready bit
5 which is set when the slave is ready to transfer data indirectly;
6 a slave mailbox register for transferring direct message data using the DMA engine if
7 the direct message ready status bit is set and with the slave mailbox registers used for transferring
8 message data using the DMA engine if an indirect message ready status bit is set;
9 a message transfer complete status interrupt at the slave that is set to indicate when
10 the transfer of the message is complete;
11 a slave indirect address mailbox register for transferring address data using the DMA
12 engine if the indirect message ready status bit is set;
13 an indirect transfer message interrupt bit at the slave which is set to initiate the
14 indirect transfer; and
15 a slave memory map used with the address data to transfer data between a location
16 indicated by the address data and the DMA engine.